

## VOLO iOS 프로젝트 설정 가이드 ver 1.0.3.20141218

### 대체 왜 Cocoapod 때문이죠?

관심사 분리

- 관심사 분리가 일어날 수밖에 없는 환경 구성
- 주변 환경, 예를 들면 관련된 코드와 독립적으로 개발 진행 가능
- 다양한 주제의 진행(=분업), 특정 주제의 체계적 진행이 가능
- 낮은 의존성 =? 테스트 용이성(testability)
  - 의존성이 있다는 것은 **의존 대상이 제대로 동작해야 한다**는 조건이 추가된다는 의미.

기능별 관리

- 프로젝트가 분리되므로 기능 분류가 체계적으로 이뤄짐
- 기능에 대한 발견, 전체 구조 살피기(Discoverability)도 용이

버전 붙이기

- 업그레이드. 새로운 버전이 나왔을 때
- 새로운 버전에 문제가 있다면? 이전 버전으로 돌려야하는데 이게...

항상 출시 가능한 제품

### 일반적인 Cocoapod 사용법

다른 cocoapods를 이용만 할 경우

1. 원하는 Xcode 프로젝트(.xcodproj)를 만들기
2. Podfile을 해당 위치에 생성.
  - 의존 라이브러리 등의 정보 기술
3. pod 명령어로 프로젝트 구조 변경
  - .xcworkspace가 생성됨
  - pod install, pod update

4. xcworkspace를 이용해 작업

cocoapods 환경에서 활용할 프로젝트를 만드는 경우

- **Volo 프로젝트 준비 과정** 이하 내용을 참고함

### Volo 프로젝트 준비 과정

Volo Pod Repo [https://github.com/WanderWorld/\\_VoloPodRepo](https://github.com/WanderWorld/_VoloPodRepo) 를 참고하여 Volo 전용 Pod 저장소를 추가

```
pod repo add volo
```

[https://github.com/WanderWorld/\\_VoloPodRepo](https://github.com/WanderWorld/_VoloPodRepo) [https://github.com/WanderWorld/\\_VoloPodRepo](https://github.com/WanderWorld/_VoloPodRepo)

## 작성 방법1: 신규 프로젝트 제작

cocoapod와 xcode를 활용하여 직접 처음부터 프로젝트를 작성하는 방법과, 제공되는 VLOSampleProject를 수정하여 사용하는 방법이 있다.

### 프로젝트 직접 생성하기

1. 접두어 **VLO**를 사용하여 프로젝트 명을 정한다.  $\{VLOProjectName\}$ 라 가정하자. - 예제에서는 VLOSampleProject라는 이름을 사용

`pod lib create  $\{VLOProjectName\}$`  <http://guides.cocoapods.org/making/using-pod-lib-create> 명령을 이용하여 기본적인 디렉토리 Layout을 작성한다.

```
pod lib create VLOSampleProject
```

- 4개의 질문이 나오는데, 다음과 같이 입력한다. 입력을 마치면 workspace가 생성된다.
  1. Demo Application을 만들겠다는 질문은 Yes(Default)로,
  2. Test Framework는 Specta(Default)로 답한다.
  3. View Based Testing은 **No**로 답한다.
  4. 접두어(prefix)는 **VLO**를 입력한다.

VLOSampleProject.podspec 파일을 적절하게 수정한다. 수정할 주된 내용은 다음과 같다.

- description
- github url
- author: 참여 멤버 정확하게 기입
- source: git 소스 위치
- platform: Volo Application은 8.0이상을 지원한다.

Demo 용 프로젝트를 새로 생성

- 목표: Example 폴더가 있던 곳과 동일한 위치에 Demo 프로젝트를 생성한다. (향후 Example 폴더는 삭제할 예정)
- Xcode를 열고, `Create a new Xcode project` 선택한다. 프로젝트명은  $\{VLOProjectName\}$ **Demo**
- 제공되는 VLOSampleProject의 경우 프로젝트명은 **VLOSampleProjectDemo**이다.
- 작성하고자 하는 프로젝트 타입을 선택한다. 제공되는 예제에서는 Single View Application이다.

Demo 프로젝트에 pod 설치

- **주의** Xcode를 열어 프로젝트 설정(target)을 확인하며 작업하는 것이

좋다.

XxxDemo/Podfile를 작성한다. Example 디렉토리에 있는 Podfile을 Demo에 복사한 후 수정한다.

```
source
'https://github.com/CocoaPods/Specs.git'

target 'VLOSampleProjectDemo', :exclusive =>
true do
  pod "VLOSampleProject", :path => "../"
end

target 'VLOSampleProjectDemoTests', :exclusive
=> true do
  pod "VLOSampleProject", :path => "../"

  pod 'Specta', '~> 0.2.1'
  pod 'Expecta'
end
```

Podfile의 주요 수정 대상은 다음과 같다.

1. target: target명은 Xcode의 프로젝트 설정을 참고한다.
2. pod: Demo 프로젝트가 의존하는 pod를 적절히 기술한다.
3. path: podspec이 위치한 경로이다.

pod install을 활용하여 pod 작업용 workspace로 변환한다.

- Podfile이 있는 위치에서 다음을 실행 > pod install  
이상 없이 workspace로 변환되었는지, 다음 메시지를 확인한다.

[!] From now on use

**VLOSampleProjectDemo.xcworkspace.**

이제, 생성된 Example 폴더를 (과감하게) 삭제한다.

```
rm -rf Example
```

Example 삭제 후 디렉토리 구조는 다음과 같다.

```
VLOSampleProject
├── .git
├── .gitignore
├── .travis.yml
├── LICENSE
├── Pod
│   ├── Assets
│   └── Classes
├── README.md
├── VLOSampleProject.podspec
└── VLOSampleProjectDemo
    ├── .DS_Store
    ├── Podfile
    ├── Podfile.lock
    └── Pods
```

```
VL0SampleProjectDemo
VL0SampleProjectDemo.xcodeproj
VL0SampleProjectDemo.xcworkspace
VL0SampleProjectDemoTests
```

XxxDemo/Pods를 .git 관리 대상에서 제외

- Xxx\*Demo/Pods\*는 Demo가 의존하는 pod 라이브러리가 저장되는 디렉토리로 버전 관리가 필요없음
- XxxDemo/Pods를 .gitignore에 추가한다.
- **주의** 필수 디렉토리인 \${VLOProjectName}/Pod를 지정하지 않도록 주의한다.

## 소스 코드 작성

작성 요령

- 결국, 라이브러리의 핵심 소스는 \${VLOProjectName}/**Pod/Classes**에 위치한다.
    - VL0SampleProject를 다운로드했다면 해당 위치에 VL0SampleClass 라는 objective-c 클래스가 있으니 참고하면 된다.
  - Xcode에서는 Pods 프로젝트에 Development Pods/\${VLOProjectName}로 표시됨
- Demo 프로젝트에서 다음과 같이 추가한 소스를 활용
1. Podfile이 있는 XxxDemo 디렉토리에서 pod update 원하는 파일에서 import 해서 사용

```
#import <VL0SampleClass.h>
```

## 작성 방법2: 제공되는 VL0SampleProject 수정하기

github에서 VL0SampleProject를 다운로드 후, pod install 수행

예를 들어, VL0StickerTextView를 만들 경우.

```
git clone
https://github.com/WanderWorld/VL0SampleProject https:
//github.com/WanderWorld/VL0SampleProject
VL0StickerTextView
```

Xcode를 바로 실행하면 Pods 프로젝트가 제외되어 있는데 Demo 프로젝트 폴더에서 pod install을 이용해 생성하면 된다.

현재 git은 VL0SampleProject로 연결되어 있으므로 .git을 삭제하여 연결을 끊는다.

podspec 변경

- mv VLOSampleProject.podspec  
VLOStickerView.podspec
- podspec의 VLOSampleProject 관련 내용 변경

위 **프로젝트 직접 생성하기**의 4~9번 과정을 참고하여 새로운 Demo 프로젝트를 생성한다.

- 단, 작성할 신규 Demo 프로젝트는 **`${VLOProjectName}Demo`**이 되고, 나중에 삭제할 대상은 Example 디렉토리가 아닌 제공되는 VLOProjectSample이 된다.
- VLOProjectSample을 이용해도 **프로젝트 직접 생성하기**의 과정을 이해하고 진행하면 도움이 된다.
- README.md 또한 적절히 수정한다.

Development Pods에 적절한 소스 파일을 추가한다.

- **주의** 결과적으로 Pod/Classes/\${VLOProjectName} 에 해당 소스가 들어있어야 Demo 프로젝트 등에서 참고 가능하다.
- 소스 추가 후 제공되는 VLOSampleClass.\*를 지운다.

pod update 후 필요시 Demo 프로젝트용 소스를 작성한다.

git을 초기화하고 github에 연결한다.

- git init 후 github에 프로젝트를 생성하고, 생성 직후 **push an existing repository from the command line** 부분의 안내에 따라 git remote 를 추가한다.
- .gitignore에 새로 작성한 XxxDemo/Pods가 제대로 지정되어있는지 확인한다.

## pod로 내보내기

사전 커밋 & tagging

**주의** podspec의 내용(버전)과 일치하도록 git tagging을 해야 향후 pod를 이용해 특정 버전의 내용을 선택해 가져올 수 있음

```
git tag -a 0.1.0 -m 'version 0.1.0'
git push origin 0.1.0
```

podspec 문법/내용 검사

```
pod lib lint
```

Volo용 pod 저장소에 podspec 추가

```
pod repo push volo ${VLOProjectName}.podspec
```

